1. A method of generating circuit simulation code using a computer language, comprising:

declaring a width of a state variable equal to a width of a vector state, the vector state having a width greater than a system platform width; and

extracting data from the vector state and placing the data in the state variable.

2. The method of claim 1, further comprising generating the vector state using a simulator console.

3. The method of claim 2, wherein generating the vector state comprises specifying the width of the vector state.

4. The method of claim 1, further comprising displaying all n ($n \geq 1$) bits of the vector state, the width being n bits wide.

5. The method of claim 1, further comprising using the state variable in a single action.

6. The method of claim 5, wherein using the state variable in a single action comprises comparing the state variable to a second state variable.

7. The method of claim 1, further comprising generating a simulation script that includes the state variable, the simulation script for driving a simulation.

8. The method of claim 1, further comprising treating the vector state as a native simulator object, the native simulator object having a maximum state size allowable by a simulator in a single action.

9. The method of claim 1, wherein the data comprises n ($n \geq 1$) bits of a simulator state.

10. The method of claim 1, wherein extracting data from the vector state comprises extracting all n ($n \geq 1$) bits of the vector state, the width being n bits wide.

11. An apparatus comprising:

a memory that stores executable instructions for generating circuit simulation code using a computer language; and

a processor that executes the instructions to:

declare a width of a state variable equal to a width of a vector state, the vector state having a width greater than a system platform width; and

extract data from the vector state and placing the data in the state variable.

12. The apparatus of claim 11, further comprising instructions to generate the vector state using a simulator console.

13. The apparatus of claim 12, wherein to generate the vector state comprises specifying the width of the vector state.

14. The apparatus of claim 11, further comprising instructions to display all n ($n \geq 1$) bits of the vector state, the width being n bits wide.

15. The apparatus of claim 11, further comprising instructions to use the state variable in a single action.

16. The apparatus of claim 15, wherein to use the state variable in a single action comprises comparing the state variable to a second state variable.

17. The apparatus of claim 11, further comprising instructions to generate a simulation script that includes the state variable, the simulation script for driving a simulation.

18. The apparatus of claim 11, further comprising instructions to treat the vector state as a native simulator object, the native simulator object having a maximum width size allowable by a simulator in a signal action.

19. The apparatus of claim 11, wherein the data comprises n ($n \geq 1$) bits of a simulator state.

20. The method of claim 11, wherein to extract the data from the vector state comprises extracting all n ($n \geq 1$) bits of the vector state, the width being n bits wide.

21. An article comprising a machine-readable medium that stores executable instructions for generating circuit simulation code using a computer language, the instructions causing a machine to:

declare a width of a state variable equal to a width of a vector state, the vector state having a width greater than a system platform width; and

extract data from the vector state and placing the data in the state variable.

22. The article of claim 21, further comprising instructions causing the machine to generate the vector state using a simulator console.

23. The article of claim 22, wherein to generate the vector state comprises specifying the width of the vector state.

24. The article of claim 21, further comprising instructions causing the machine to display all n ($n \geq 1$) bits of the vector state, the width being n bits wide.

25. The article of claim 21, further comprising instructions causing the machine to use the state variable in a single action.

26. The article of claim 25, wherein to use the state variable in a single action comprises comparing the state variable to a second state variable.

27. The article of claim 21, further comprising instructions causing the machine to generate a simulation script that includes the state variable, the simulation script for driving a simulation.

28. The article of claim 21, further comprising instructions causing the machine to treat the vector state as a native simulator object, the native simulator object having a maximum width size allowable by a simulator in a signal action.

29. The article of claim 21, wherein the data comprises n ($n \geq 1$) bits of a simulator state.

30. The article of claim 21, wherein to extract the data from the vector state comprises extracting all n ($n \geq 1$) bits of the vector state, the width being n bits wide.

31. A computer instruction that generates a vector state, the vector state having a width larger than a predefined state width.

32. The computer instruction of claim 31, wherein the vector state is generated using a simulator console.

33. The computer instruction of claim 31, wherein the vector state is used to generate state variables.